REMARKS

Applicant thanks the Examiner for a thorough search of the present application, but respectfully requests reconsideration of the present application in view of the reasons that follow.

Claims 35 and 36 are requested to be cancelled.

Claims 27 and 31-34 are currently being amended.

Claims 37 and 38 are new claims.

This amendment adds, changes and/or deletes claims in this application. A detailed listing of all claims that are, or were, in the application, irrespective of whether the claim(s) remain under examination in the application, is presented, with an appropriate defined status identifier.

After amending the claims as set forth above, claims 1-34, 37, and 38 are now pending in this application.

I. **Rejection of claims 27, 33, and 34 under 35 U.S.C. § 101**

In the outstanding Office Action of February 7, 2008, the Examiner rejected claims 27, 33, and 34 under 35 U.S.C. § 101 because the claims appeared to be "comprised of software alone without claiming associated computer hardware required for execution." In response to this rejection, Applicant has amended claims 27, 33, and 34 to recite that the software or interface module is embodied on a computer-readable medium. An example of a computer-readable medium is the "memory" described in paragraphs [0059], [0060], and [0076] of the present application. Applicant therefore submits that the amendment is fully supported by the application as originally filed. As such, Applicant submits that claims 27, 33, and 34 as amended are directed to statutory subject matter and respectfully requests that the rejection be removed.

II. Rejection of claims 31-36 under 35 U.S.C. § 112, 2nd paragraph

On page 2 of the Office Action, the Examiner rejected claims 31-36 under 35 U.S.C. § 112, 2nd paragraph because, in the Examiner's opinion, the "claim languages are unclear and indefinite." In particular, the Examiner asserted that "it is uncertain if these claims are system claims or method claims." In response to these rejections, Applicant has amended claims 31-34 to independent form. In addition, Applicant has cancelled claims 35 and 36. Accordingly, Applicant submits that the claims as amended are clear and definite. Thus, Applicant respectfully requests withdrawal of the rejection.

III. Rejection of claims 1, 4, 8-10, 12, 15-17, 19, 21, 24, and 26-36 under 35 U.S.C. § 102(e)

On page 3 of the Office Action, the Examiner rejected claims 1, 4, 8-10, 12, 15-17, 19, 21, 24, and 26-36 under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 7,031,989 to Elmendorf et al. (Elmendorf). Applicant respectfully traverses the rejection for the reasons set forth below.

The Examiner asserted that Elmendorf teaches all of the required limitations of at least independent claims 1, 15, 27, and 28. Applicant respectfully disagrees with the Examiner's position. In particular, Applicant submits that the Examiner failed to make a prima facie case of anticipation. For a prior art reference to anticipate the claim of a patent, the reference must disclose each and every limitation of a claimed invention. *See Apple Computer, Inc. v. Articulate Systems, Inc.,* 234 F.3d 14, 20 (Fed. Cir. 2000). In this instance, the Examiner has not shown where each and every limitation of independent claims 1, 15, 27, and 28 is taught, suggested, or inherent in Elmendorf. In particular, Applicant submits that (1) the Examiner improperly rejected independent claims 1, 15, 27, and 28 as a group, (2) the Examiner improperly rejected independent claims 1, 15, 27, and 28 by applying two mutually exclusive embodiments described in Elmendorf, and (3) none of the embodiments, either separately or in combination, read on independent claims 1, 15, 27, and 28.

Applicant submits that the rejection of independent claims 1, 15, 27, and 28 as a single group is improper. In rejecting these claims, the Examiner quoted and addressed the text of only claim 1. However, Applicant submits that each claim recites different relevant

features that cannot properly be rejected by only addressing claim 1. Doing so, fails to enable the Applicant to properly reply to the merits of the rejection of independent claims 15, 27, and 28. As such, Applicant respectfully requests the Examiner to address independent claims 15, 27, and 28 separately on each of their respective merits.

In addition, Applicant submits that the rejection is also improper because the Examiner cited steps from two mutually exclusive embodiments discussed in Elmendorf. Specifically, Elmendorf describes two very different embodiments related to different situations that even if combined, would not read on the claims because the result would not be an operable system. The first embodiment is described at Col. 5 line, 1 – Col. 7, line 53 and relates to situations were the data object is being *replaced* via duplication and involves the use a system freshness indicator and also specific freshness indicators for each context. (*see, e.g.,* Figures 6-9). The second embodiment is described at Col. 7, line 55 – Col. 9, line 40 and relates to situations were the data object is being *modified* and involves the use of a single system usage counter and includes temporarily blocking the data. As described in greater detail below, neither embodiment, either separately or in combination, reads on the present application.

Regarding the first embodiment, Elmendorf teaches:

> When a new module (which is to share data with previously loaded modules) is loaded, a new shared data object is created (step 810). The system freshness indicator is incremented to reflect creation of the new data object (step 830). The old data object is placed on the "garbage list" and stamped with the current system freshness (step 820). <u>At this point, the new shared data object is available for access by a context</u>. Accordingly, if a context performs a new access of shared data, the new data object should be accessed instead of the old data object. Thus, when a context accesses the shared data (step 840), the freshness indicator for that context is set equal to the system freshness (step 850). <u>When all the contexts have at least caught up</u> to the freshness stamp of a given old data object (step 855), that <u>old data object is deleted</u> (step 860). (Col. 6, lines 46-60; emphasis added).

Thus, the first embodiment of Elmendorf is directed to a method for adding a new shared data object and deleting an old shared data object once each context has "caught up" with the data within the new shared data object. In particular, Elmendorf teaches that a specific freshness indicator is associated with each context and every specific freshness indicator must be equal to the system freshness indicator before the old shared data object is deleted. (*See, e.g.,* Fig. 9). Such a teaching fails to read on claim 1 of the present application because claim 1 recites in part:

> setting a replace module variable;
> when the replace module variable is set:
> a. blocking threads from entering the implementation module;
> and
> b. when all the private variables are in a reset state, replacing
> the implementation module

(Claim 1; emphasis added). Applicant submits that Elmendorf fails to teach or even suggest "setting a replace module variable" and replacing the module only after conditions are satisfied. Instead, the method of Elmendorf enables a module to be added and <u>instantly available</u> for access without satisfying preconditions such as "setting a replace module variable" and <u>waiting</u> until "all private variable are in a reset state." In other words, the present application requires "setting a replace module variable" and <u>waiting</u> until "all private variable are in a reset state" *before* replacing the implementation module. In contrast, Elmendorf replaces the module and *later* attempts to satisfy conditions related to the deletion of the old shared data object. Since Elmendorf teaches that the old shared data object is replaced at the very beginning of the method and the new shared data object is given instant access, Applicant submits that such a teaching cannot read on independent claim 1, because claim 1 requires replacing the implementation module only after satisfying numerous conditions that are not taught or even suggested in Elmendorf. One of ordinary skill in the art would recognize that such a teaching would fail to ensure that the threads are synchronized so that no thread is within the component when it is replaced. Accordingly, Applicant submits that the Examiner failed to make a prima facie case of anticipation with regard to claim 1.

With regard to the second embodiment of Elmendorf, Elmendorf teaches temporarily blocking access to a module while the module is modified. (Col. 7, lines 60-63). In order to

-13-

ensure that module is not being currently accessed, Elmendorf teaches the use of a "usage counter" that is incremented/decremented based on the number of contexts accessing the module. Once the usage counter indicates that all the contexts have exited the module, the modifications are made to the module. (*see, e.g.,* Col. 8 lines 44- 63 and Figs. 10-12).

While such a teaching relates to blocking threads from entering a module and replacing a module when all the threads have exited, Applicant submits that Elmendorf fails to teach "creating a plurality of private variables corresponding to the plurality of threads," as described in claim 1. Instead, Elmendorf teaches a global or system usage counter that monitors the entering and exiting of threads. There is no variable associated with each thread. In fact, Elmendorf teaches away from associating a variable with each thread by stating "[i]t will be appreciated by those skilled in the art that this usage blocking method <u>does not require the operating system to save the state of the executing thread </u>while the access counter is updated." (Col. 9, lines 7-12; emphasis added).

For at least the reasons discussed above, Applicant submits that Elmendorf fails to anticipate independent claim 1. Furthermore, Applicant submits that two different embodiments cannot be properly combined because doing so would produce an inoperable result. This is because the first embodiment requires instant access to the new module and deletion of the old module. In contrast, the second embodiment requires delayed access to the new module and <u>not</u> deleting the old module. Accordingly, the combination of the two embodiments would not be logical and would produce and inoperable result. As such, Applicant respectfully requests withdrawal of the rejection based upon Elmendorf.

Furthermore, independent claim 15 recites in part:

> when a thread is within the portion of code and the perform
> action variable is set, the thread:
> a. resetting the private variable;
> b. when the private variables for all threads are not set:
> > i. performing the action; and
> > ii. resetting the perform action variable;
> c. setting the private variable; and
> when a thread is within the portion of code and the perform
> action variable is reset, the thread:

-14-

> d. using the resource; and
> e. resetting the private variable; wherein the threads may be
> dynamically created and destroyed.

In addition independent claim 27 recites in part:

> each private variable to be readable by all threads and writable
> only by the corresponding thread, and each private variable
> arranged to be in a SET state or a RESET state; ...
>
> program code arranged for registering the privates variables for
> created threads, deregistering the private variables for destroyed
> threads, ... replacing the implementation module when all the
> registered private variables are in a RESET state;

Additionally, independent claim 28 recites in part:

> a processor arranged for registered the private variables for
> created threads, deregistering privates variables for destroyed
> threads, setting and resetting the registered private variables
> when instructed by the corresponding thread, setting the replace
> module variable

Applicant submits that the rejection of each of the above claims is improper because the above claim elements were not addressed in the outstanding Office Action. Because the pending Elmendorf rejection of independent claims 15, 27, and 28 has failed altogether to address the above-quoted features, Applicant is presently prevented from developing with the Examiner a clear issue regarding the relationship between Elmendorf and independent claims 15, 27, and 28. According to MPEP § 706.07, final rejection is improper until such a clear issue can be developed. Applicants would therefore point out that, if the next Office Action again presents a rejection of independent claims 15, 27, and 28 based on Elmendorf, including an attempt to cure the aforementioned deficiency in the pending Elmendorf rejection, then that Office Action cannot properly be final.

IV.   Rejection of Claims 2, 3, 5-7, 11, 13, 14, 18, 20, 22, 23, and 25 under 35 U.S.C. § 103(a)

On page 6 of the Office Action, the Examiner rejected claims 2, 3, 5-7, 11, 13, 14, 18, 20, 22, 23, and 25 under 35 U.S.C. § 103(a) as being unpatentable over Elmendorf. Applicant respectfully traverses the rejection for the reason set forth below.

-15-

First, Applicant notes that each of these claims is dependent upon the independent claims discussed above, and Applicant therefore incorporates its arguments from above. Additionally, Applicant notes that in rejecting each claim, the Examiner noted a deficiency and asserted it would have been obvious to one of ordinary skill in the art to include the deficient feature. Thus, the Examiner is seemingly taking Official Notice without explicitly stating Official Notice. In response to these rejections, Applicant respectfully requests the Examiner to provide evidence to support that assertions that one of ordinary skill in the art would have been motivated to include the numerous features in Elmendorf. Without such evidence, the Examiner's assertion lacks "instant and unquestionable demonstration as being well-known," as required by MPEP § 2144.03.

V.   Conclusion

Because the reference cited by the Examiner fails to teach all of the required limitations of independent claims 1, 15, 27, 28, and 31-34, Applicant submits that each of these independent claims are patentable over this prior art. Furthermore, because dependent claims 2-14, 16-26, 29-34, 37, and 38 are each directly or indirectly dependent upon independent claims 1, 15, 27, and 28, Applicant submits that each of these claims are allowable for at least the same reasons as discussed above.

Applicant believes that the present application is now in condition for allowance. Favorable reconsideration of the application as amended is respectfully requested.

The Examiner is invited to contact the undersigned by telephone if it is felt that a telephone interview would advance the prosecution of the present application.
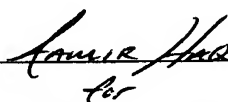
The Commissioner is hereby authorized to charge any additional fees which may be required regarding this application under 37 C.F.R. §§ 1.16-1.17, or credit any overpayment, to Deposit Account No. 08-2025. Should no proper payment be enclosed herewith, as by a check or credit card payment form being in the wrong amount, unsigned, post-dated, otherwise improper or informal or even entirely missing, the Commissioner is authorized to charge the unpaid amount to Deposit Account No. 08-2025. If any extensions of time are needed for timely acceptance of papers submitted herewith, Applicant hereby petitions for

such extension under 37 C.F.R. §1.136 and authorizes payment of any such extensions fees to Deposit Account No. 08-2025.

Respectfully submitted,

Date ___3/28/08___

By _____ 61,052

HEWLETT-PACKARD CO.
Customer Number: 22879

William T. Ellis
Attorney for Applicant
Registration No. 26,874
Telephone: (202) 672-5485

-17-